

# State space; Canonical Forms

```
%{  
num = 1;  
den = [1 18 80];  
H = tf(num, den);  
zpk(H)  
[A, B, C, D] = tf2ss(num, den)  
%}
```

```
m=7; n=2; z = -[m,m+2,m+3]; p = -[n,n+2,n+3,n+4];  
%aducem functia in forma zero-pole-gain  
H = zpk(z,p,1)
```

```
H =  
  
      (s+7) (s+9) (s+10)  
      -----  
      (s+2) (s+4) (s+5) (s+6)
```

Continuous-time zero/pole/gain model.

```
%returnam coeficientii polinoamelor de la numarator si numitor din functia  
%de transfer in forma unui vector  
[num,den] = tfdata(H, 'v')
```

```
num = 1x5  
      0      1      26      223      630  
den = 1x5  
      1      17     104     268     240
```

```
%aducem functia de transfer in forma a spatiului starilor pentru a obtine  
%forma canonica de control  
[Ac, Bc, Cc, Dc] = tf2ss(num, den)
```

```
Ac = 4x4  
    -17    -104    -268    -240  
      1         0         0         0  
      0         1         0         0  
      0         0         1         0  
Bc = 4x1  
      1  
      0  
      0  
      0  
Cc = 1x4  
      1      26      223      630  
Dc = 0
```

```
%A este matricea sistem si este de dimensiunea 4x4, pentru ca avem 4  
%variabile de stare  
%B este matricea de intrare si este de forma 4x1 pentru ca sistemul are 1 input
```

```
%C este matricea de iesire de forma 1x4 pentru ca sistemul are 1 output  
%D arata transferul direct care este 0
```

```
[Ac, Bc, Cc, Dc] = ccf(num,den);
```

```
function [A,B,C,D] = ccf(num,den)  
    sz = length(den)-1;  
    A = [-den(2:end);[eye(sz-1),zeros(sz-1,1)]];  
    B = [1;zeros(sz-1,1)];  
    C = [zeros(1,(sz-length(num))) num];  
    D = 0;  
end
```